Элементы теории алгоритмов

Важные математические проблемы имеют вид:

для некоторого данного множества X найти эффективную процедуру (т.е. алгоритм), с помощью которой можно для каждого элемента x этого множества X определить за конечное число шагов, будет этот элемент обладать некоторым данным свойством P или нет (т.е. $x \in P^+$ или $x \notin P^+$).

Решением такой проблемы является построение и обоснование искомого алгоритма.

Массовые задачи – задачи распознавания и оптимизации.

Примеры массовых задач:

- СУМ – задача сложения целых чисел.
- ДЕЛ – задача делимости целых чисел.
- НОД – задача нахождения наибольшего общего делителя двух целых чисел.
- ВЫП (SAT) – задача выполнимости формулы алгебры высказываний.
- ГП – задача существования гамильтонова пути.
- НМ – задача о независимом множествею

Под *алгоритмом* понимается совокупность инструкций о том, как решить некоторую массовую задачу.

Общие свойства алгоритма:

1)*дискретность алгоритма*;

2)*детерминированность алгоритма*;

3)*элементарность шагов алгоритма*;

4)*массовость алгоритма*.

Так как конструктивные объекты можно кодировать словами конечного алфавита Σ (например, состоящего из двоичных символов 0 и 1), то алгоритм моделируется устройством, перерабатывающим слова алфавита Σ.

Понятие алгоритма имеет смысл лишь в том случае, если множество его возможных исходных данных является потенциально обозримым множеством, которое состоит из последовательно конструируемых объектов.

Примеры: N и Σ^* .

Далее при изучении алгоритмов ***A*** будем предполагать, что множество рассматриваемых объектов X, область применения алгоритма D_A и множество возможных значений алгоритма Y являются потенциально обозримым бесконечными множествами последовательно конструируемых объектов.

Аксиома: для любых двух таких множеств X, Y существует вычислимая биекция X на Y.

Тезис Черча:

класс задач, решаемых в любой формальной модели алгоритма, совпадает с классом задач, которые могут быть решены интуитивно эффективными вычислениями, т.е. алгоритмическими методами.

Алгоритмически неразрешимые задачи и необходимость строго математического определения алгоритма.

Модели алгоритма:

1) понятие *рекурсивной функции*, введенное Клини в 1936 г.,
2) понятие *машины Тьюринга*, введенное Постом и Тьюрингом в 1936 г.,
3) понятие *нормального алгорифма*, введенное Марковым в 1954 г.,
4) понятии *формальной грамматики*, введенное Хомским в 1957 г.

Рекурсивные функции

Пусть N – множество неотрицательных целых чисел и $\boldsymbol{F}$ – множество всех частичных функций нескольких числовых переменных из N со значениями в $\boldsymbol{N}$.

0-*функция*, *функция следования* и n–местная *функция проекции* на m–ую координату определяются по формулам:

$$o(x) = 0,\ s(x) = x + 1 \text{ и } I_m^n(x_1, \dots, x_n) = x_m$$

для любых значений $x, x_1, \dots, x_n \in N$.

Функции $o(x)$, $s(x)$, I_m^n называются также *простейшими примитивно рекурсивными функциями*.

На множестве $\boldsymbol{F}$ рассмотрим следующих два оператора:

– *оператор суперпозиции* S ставит в соответствие каждой функции m переменных $f \in \boldsymbol{F}$ и m функциям n переменных $g_1, \dots, g_m \in \boldsymbol{F}$ функцию n переменных $h = S(f, g_1, \dots, g_m)$, определяемую равенством:

$$h(x_1, \dots, x_n) = f(g_1(x_1, \dots, x_n), \dots, g_m(x_1, \dots, x_n));$$

– *оператор примитивной рекурсии* R ставит в соответствие каждой функции $n+2$ переменных $f \in \boldsymbol{F}$ и функции n переменных $g \in \boldsymbol{F}$ функцию $n+1$ переменных $h = R(f, g)$, удовлетворяющую следующей *схеме примитивной рекурсии:*

$$\begin{cases} h(x_1,\ldots,x_n,0) = g(x_1,\ldots,x_n), \\ h(x_1,\ldots,x_n,y+1) = f(x_1,\ldots,x_n,y,h(x_1,\ldots,x_n,y)). \end{cases}$$

В частности, при $n = 0$ схема примитивной рекурсии имеет следующий вид:

$$\begin{cases} h(0) = a, \\ h(y+1) = f(y,h(y)), \end{cases}$$

где a — постоянная 0-местная функция, равная числу a.

<u>Определение</u>.

Функция $f \in \boldsymbol{F}$ называется *примитивно рекурсивной* (сокращенно ПРФ), если существует последовательность функций $f_1, \ldots, f_n \in \boldsymbol{F}$, в которой $f_n = f$ и всякая функция f_i является простейшей ПРФ или получается из предыдущих функций с помощью оператора суперпозиции S или оператора примитивной рекурсии R.

Примеры.

1. Функция *сложения* $x+y$ примитивно рекурсивна в силу схемы примитивной рекурсии:

$$\begin{cases} x+0=I_1^1(x), \\ x+(y+1)=s(x+y). \end{cases}$$

2. Функция *умножения* $x\cdot y$ примитивно рекурсивна в силу схемы примитивной рекурсии:

$$\begin{cases} x\cdot 0=o(x), \\ x\cdot(y+1)=x\cdot y+x. \end{cases}$$

На множестве $\boldsymbol{F}$ всех частичных функций нескольких числовых переменных рассмотрим еще один оператор μ, который называется *оператором ограниченной минимизации*: каждой функции m переменных $g \in \boldsymbol{F}$ оператор μ ставит в соответствие функцию m переменных $f=\mu(g)$, определяемую равенством:

$$f(x_1,\ldots,x_m) = \mu_y(g_1(x_1,\ldots,x_{m-1},y) = x_m),$$

где правая часть равенства обозначает наименьшее решение уравнения $g_1(x_1,\ldots,x_{m-1},y) = x_m$ относительно y.

<u>Определение</u>. **Функция** $f \in \boldsymbol{F}$ называется *частично рекурсивной* (сокращенно **ЧРФ**), если существует последовательность функций $f_1, \ldots, f_n \in \boldsymbol{F}$, в которой $f_n = f$ и всякая функция f_i является простейшей **ПРФ** или получается из предыдущих функций с помощью оператора суперпозиции S, оператора примитивной рекурсии R или оператора ограниченной минимизации μ.

При этом частично рекурсивная функция называется *рекурсивной* (сокращенно **РФ**), если она всюду определена.

Множества всех рекурсивных, частично рекурсивных и примитивно рекурсивных функций из множества F обозначим соответственно $F_{РФ}$, $F_{ЧРФ}$ и $F_{ПРФ}$.

Из определений следуют включения множеств:

$$F_{ПРФ} \subset F_{РФ} \subset F_{ЧРФ}.$$

Известно, что все включения являются собственными, т.е.

$$F_{ПРФ} \neq F_{РФ} \neq F_{ЧРФ}.$$

Понятия рекурсивной функции и частично рекурсивной функции естественно переносятся на словарные функции и языки над любым конечным алфавитом $A = \{a_0, \ldots, a_{n-1}\}$ с помощью биекции λ множества A^* на множество $\boldsymbol{N}$, которая определяется по правилам:

$$\lambda(\Lambda) = 0 \text{ и } \lambda(w) = i_k n^k + i_{k-1} n^{k-1} + \ldots + i_1 n + i_0$$

для непустого слова $w = a_{i_k} a_{i_{k-1}} \ldots a_{i_1} a_{i_0}$.

Такая биекция λ называется *лексикографической функцией* для алфавита A.

Лексикографическая функция λ естественно продолжается на множество всех частичных словарных функций $f:\left(A^*\right)^n \to A^*$ по правилу:

$\lambda(f)$ есть числовая функция из $\boldsymbol{N}^n$ со значениями в $\boldsymbol{N}$, которая определяется уравнением

$$\lambda(f)\left(x_1,\ldots,x_n\right)=\lambda\left(f\left(\lambda^{-1}(x_1),\ldots,\lambda^{-1}(x_n)\right)\right)$$

для значений $\left(x_1,\ldots,x_n\right)\in \boldsymbol{N}^n$.

Определение.

Частичная словарная функция $f:\left(A^*\right)^n \to A^*$ называется частично рекурсивной (соответственно, рекурсивной или примитивно рекурсивной), если числовая функция $\lambda(f)$ частично рекурсивна (соответственно, рекурсивна или примитивно рекурсивна).